Yale University
Department of Computer Science

**Contour-Based Binocular Stereo: Inferring
Coherence in Stereo Tangent Space**

Gang Li   Steven W. Zucker
Yale University   Yale University

Contour-Based Binocular Stereo: Inferring Coherence in Stereo Tangent Space

Gang Li

Yale University

Steven W. Zucker

Yale University

Abstract

Standard approaches to stereo correspondence have difficulty when scene structure does not lie in or near the frontal-parallel plane, in part because an orientation disparity as well as a positional disparity is introduced. We propose a correspondence algorithm based on differential geometry, and inspired by neurobiology, that takes explicit advantage of both disparities. The algorithm relates the 2D differential structure (position, tangent, and curvature) of curves in the left and right images to the Frenet approximation of the (3D) space curve. A compatibility function is defined via transport of the Frenet frames, and they are matched by relaxing this compatibility function on overlapping neighborhoods along the curve. The remaining false matches are concurrently eliminated by a model of "near" and "far" neurons derived from neurobiology. Examples on scenes with complex 3D structures are provided.

1   Introduction

Stereo vision has made significant progress in computer vision [2, 11, 36, 4]. Nevertheless, Fig. 1 reveals many of the problems still remaining for stereo correspondence algorithms: structure is not fronto-parallel, as in many man-made scenes; ordering constraints [33] are violated accordingly; point-wise uniqueness constraints [23, 33, 41] are inconsistent with self-occlusion; branching and discontinuities are inconsistent with smoothness constraints [23, 26]; and feature density is not uniform. Nevertheless, there is a structural relationship between features in the (left, right) image pair and the 3D scene, and our goal in this paper is to develop this relationship into a new stereo-correspondence algorithm.

There are two sources of motivation for our algorithm. The first is biological: tree-dwelling primates are spectacularly adept at stereo [34], which

1

implies they can solve the problems in Fig. 1. The neurophysiology in support of this is classical; and begins with binocularly-selective neurons that are also orientation selective[17, 32]. This suggests an approach based on differential geometry, with orientation identified with the tangent to spatial structure, and leads to our second motivation: to use differential geometry [12] to derive the underlying structural relationship between image features and 3D spatial structure. But even with the local geometric constraint there still exist ambiguities: different pairings of image features give rise to different 3D structures, thereby creating multiple ghost matches as well as correct ones. So finally we return to the biological motivation, and introduce an extended model for binocular neurons that implements their "near" and "far" tuning properties to eliminate these ghost matches. The result is a system that works on a wide range of imagery rich in contours that meander through space.

In our algorithm we first need to compute the space curve local approximations based on image measurements. Our geometric calculations follow the literature on curve matching. Cipolla and Zisserman [8] determined image tangent and geodesic curvature from a single view of a space curve under perspective projection. Faugeras and Robert [35, 14] used tangent and curvature constraints in a trinocular system, and predicted the curvature at a point in the third image based on measurements in the first two. Assuming knowledge of the fundamental matrix, Schmid and Zisserman [37] described how to compute the normal at one 3D space curve point, thus determining the osculating plane (in the form of its planar homography) at that point from corresponding tangents and curvatures at two perspective image points. We extend this previous work by showing how to compute both the normal and the curvature at one 3D space curve point from the perspective projections of the space curve in two views.

With the space curve local approximations at hand, we futher require neighboring matching pairs to be consistent. Since the original stereo correspondence problem cannot be solved by the point-wise geometric constraints, this idea is natural [24, 25, 33, 29, 6, 41, 38]. But the question of how to define and formulate good measures of the consistency between two nearby matching pairs has not been well answered. An early cooperative algorithm[24, 25] simply uses a local excitatory neighborhood of the same disparity level to support the current matching pair. In the PMF algorithm [33] each potential match receives support from all potential matches in its neighborhood that satisfy the disparity gradient limit. In the refined cooperative algorithm [41] the local support is defined as the sum of all match values within a 3D local support area, designed to include a small

disparity range as an extra dimension. Shan and Zhang [38] use combined binary measurements for line segments and curves as the compatibility function.

We argue that the neighborhood support information, or the compatibility measure between two nearby possible matches, can be derived geometrically. The normals and curvatures provide the essentials for the basis of our algorithmic framework, which follows the relaxation or belief propagation model. The compatibility fields are defined from a local (Frenet) approximation to a space curve, and these are used to determine how compatible two pairs of candidate matches are. In a first attempt the space curve was heuristically approximated [1]. In this work we individually compute the space curve local approximation for each candidate match pair using curvature. The result is a system that operates on position and orientation disparities concurrently while integrating the near and far neuron mechanism to address the remaining ambiguity. Although orientation disparity is widely studied in psychophysics [16], to our knowledge this is the first time such mechanisms have been used together in stereo matching.

1.1 Overview of Our Approach

Our correspondence algorithm is based on differential geometry and takes explicit advantage of both position and orientation disparities. The basic idea is as follows (Fig. 2). A curve in $\mathbb{R}^3$ has a tangent, normal, and binormal frame (Frenet frame) associated with every regular point along it. For simplicity, consider only the tangent in this frame, and imagine it as an (infinitly) short line segment. This space tangent projects into a planar tangent in the left image, and a planar tangent in the right image. Thus, space tangents project to pairs of image tangents. Now, consider the next point along the space curve; it too has a tangent, which projects to another pair of image tangents, one in the left image and one in the right image. The key concept that we utilize in this paper is *transport*, or the movement of the frame in $\mathbb{R}^3$ from the second point back to the first, which is essentially contextual information expressed geometrically; note that this transport has a correspondence in the left-right image pairs. Our goal is to use this transport to find corresponding pairs of image tangents such that their image properties match, as closely as the geometry can be approximated, the actual space tangents. Two notions of disparity arise from the above transport model. First, the standard notion of positional disparity corresponds, through the camera model, to depth. Second, an orientation disparity is introduced if the space tangent is not in the epipolar plane. In
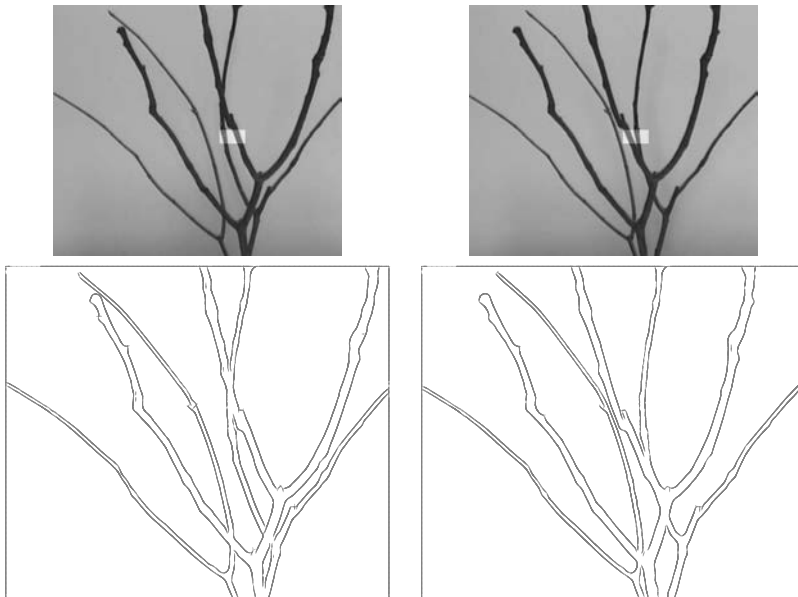
3

Figure 1: (top) Natural scenes are dense in physical structure with depth discontinuities, occlusions, and junctions, thereby posing a problem to traditional stereo algorithms. For the region around the center of the left image, when compared to its corresponding part in the right image, note the left-right ordering constraint fails inside the window because of the depth discontinuity. Partial occlusion at the junction would cause a problem for the point-wise uniquess constraint. Our algorithm is designed to function on image pairs such as this. (bottom) The (left,right) tangent (edge) map pair for the (top) images. Such tangent maps form the basis for our correspondence algorithm, and suggest a curve-based approach.

the computational vision literature, orientation disparity is largely unexplored, except in special cases [20, 39]. The success of our system derives, in part, from the simultaneous use of position and orientation disparities, and the underlying differential geometry that naturally combines them.

2 Geometry of Space Curves

Let $\alpha(s)$ be a regular curve parametrized by arc length in $\mathbb{R}^3$. If the Frenet (tangent, normal, binormal) frame $\{\mathbf{T}_0, \mathbf{N}_0, \mathbf{B}_0\}$, the curvature κ_0, and torsion τ_0 are known at $\alpha(0)$(w.l.o.g.), we can obtain a local *Frenet approximation* of the curve by taking the third-order Taylor expansion of α at $s = 0$ and keeping only the dominant terms: $\hat{\alpha}(s) = \alpha(0) + s\mathbf{T}_0 + \frac{s^2}{2}\kappa_0\mathbf{N}_0 + \frac{s^3}{6}\kappa_0\tau_0\mathbf{B}_0$.

The projection operator π maps this (local) Frenet frame to the left and right image planes:
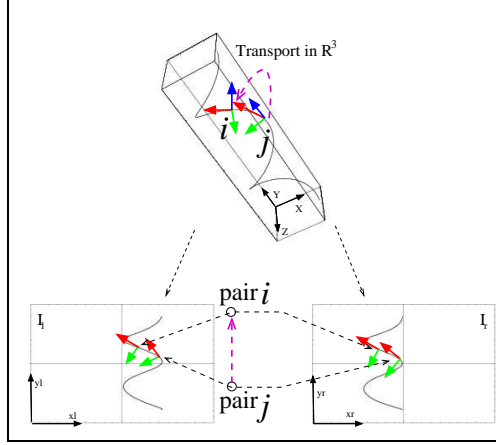
Figure 2: **Transport** of the Frenet frames in $\mathbb{R}^3$ helps finding consistent left-right image tangent pairs. Position disparity and orientation disparity are combined naturally.

$$\pi : \mathbb{R}^3 \times T(\mathbb{R}^3) \times N(\mathbb{R}^3) \times B(\mathbb{R}^3) \times \kappa(\mathbb{R}) \times \tau(\mathbb{R}) \mapsto$$
$$\mathbb{R}^2 \times T(\mathbb{R}^2) \times \kappa(\mathbb{R}) \times \mathbb{R}^2 \times T(\mathbb{R}^2) \times \kappa(\mathbb{R})$$

That is, the Frenet trihedron in $\mathbb{R}^3$ projects to two Frenet (tangent, normal)-dihedra, one in the left image and one in the right image, each augmented by curvature. Notation $T(\mathbb{R}^3)$ denotes the tangent space to $\mathbb{R}^3$; while this is isomorphic to $\mathbb{R}^3$ our notation is chosen to facilitate the reader's intuition. We refer to the space $\mathbb{R}^2 \times \mathbb{R}^2 \times T(\mathbb{R}^2) \times T(\mathbb{R}^2) \times \kappa(\mathbb{R}) \times \kappa(\mathbb{R})$ as the *stereo tangent space*; a point in this space is $(x_l, y_l, x_r, y_r, \theta_l, \theta_r, \kappa_l, \kappa_r)$, where x_i and y_i are the image projection coordinates of $\mathbf{X} \in \mathbb{R}^3$, θ_i the orientation of projected tangent in the image planes, and κ_i the image curvatures, with $i = \{l, r\}$ representing left and right images.

2.1 3D Space Curve Structure From Two Views

We now seek to determine the inverse mapping π^{-1}, the space curve structure around the 3D point $\mathbf{X}$ from its projection in the stereo tangent space. Unfortunately, as we will show, π^{-1} is not one-to-one: given a node in the stereo tangent space $i = (x_l, y_l, x_r, y_r, \theta_l, \theta_r, \kappa_l, \kappa_r)$, we can only determine the position $\mathbf{X}$, the Frenet frame $\{\mathbf{T}, \mathbf{N}, \mathbf{B}\}$, and the curvature κ at the space curve point. The torsion τ can not be determined. To prove this we assume the following image measurements are given: position, tangent,

5

curvature, i.e. (x, y, θ, κ) at every image curve point in both images. We further suppose (for space reasons)that the 3D position $\mathbf{X}$ and tangent $\mathbf{T}$ are computed by standard methods [13, 15]. Note that in this article we assume camera calibration information is available. We now describe how to compute the normal and curvature at a 3D space curve point from two views.

We begin with a standard construction [7]. Let α be a smooth space curve with non-vanishing curvature κ. Denote its position by the vector $\mathbf{r}(s)$ in the world coordinate system, and its spherical projection to the unit radius image sphere with center $\mathbf{c}$ by a vector $\mathbf{u}$ in the camera coordinate system (see Fig. 3(a)). Assume the world and the camera coordinate systems have the same orientation, that is, there is only translation (no rotations) between them. Then the space curve can be described as: $\mathbf{r}(s) = \mathbf{c} + \lambda(s)\mathbf{u}(s)$, where $\mathbf{c}$ is the vector pointing to the camera center in the world coordinate system, λ is the Euclidean distance of $\mathbf{r}(s)$ to the camera coordinate system origin, and $\mathbf{u}$ is the vector pointing to its spherical projection in the camera coordinate system.

Previous work [8, 7] shows that the relationship of image geodesic curvature and the 3D space curve geometry under perspective projection is:

$$\kappa^g = \frac{\lambda\kappa(\mathbf{u} \times \mathbf{T}) \cdot \mathbf{N}}{(1 - (\mathbf{u} \cdot \mathbf{T})^2)^{\frac{3}{2}}} \qquad (1)$$

where κ^g is the image geodesic curvature at the image spherical projection $\mathbf{u}$ of the space point being studied, $\mathbf{T}$ and $\mathbf{N}$ are the space tangent and normal at that point, κ is the curvature at the space point $\mathbf{r}(s)$. We now calculate the relationship between 3D space curve properties and their (measurable) perspective projection image plane properties.

Consider the 2D image plane curve formed by $\mathbf{r}(s)$ through perspective projection, with $\mathbf{p}$ the vector pointing to the projection in the image plane ($\mathbf{u} = \mathbf{p}/\|\mathbf{p}\|$), and $\mathbf{f}$ the vector pointing to the image center, both in the camera coordinate system. Since we choose the camera and image coordinate systems such that they are aligned, two dimensional vectors tangent and normal in the image coordinate system can be lifted to $\mathbb{R}^3$, the camera coordinate system, to $\mathbf{t}$ and $\mathbf{n}$, with the third component being zero. Note that this construction will form the same spherical projection as $\mathbf{r}(s)$. Furthermore, although for a plane curve the curvature is defined as the signed curvature κ^p, and the normal is defined such that the tangent-normal basis has the same orientation as the coordinate axis basis, the quantity $\kappa^p\mathbf{n}$ is still the same if we study this curve as a 3D curve. The binormal at any

point on this curve is $\mathbf{b} = \mathbf{t} \times \mathbf{n} = \mathbf{f}/\|\mathbf{f}\|$. The relationship between the perspective projection image curvature and the geodesic curvature is thus given by:

$$
\begin{aligned}
\kappa^g &= \frac{\|\mathbf{p}\|\kappa^p(\mathbf{u} \times \mathbf{t}) \cdot \mathbf{n}}{(1 - (\mathbf{u} \cdot \mathbf{t})^2)^{\frac{3}{2}}} & &= \frac{\|\mathbf{p}\|\kappa^p\mathbf{u} \cdot (\mathbf{t} \times \mathbf{n})}{(1 - (\mathbf{u} \cdot \mathbf{t})^2)^{\frac{3}{2}}} \\
&= \frac{\|\mathbf{p}\|\kappa^p\mathbf{u} \cdot \frac{\mathbf{f}}{\|\mathbf{f}\|}}{(1 - (\mathbf{u} \cdot \mathbf{t})^2)^{\frac{3}{2}}} & &= \frac{\frac{1}{\|\mathbf{f}\|}\kappa^p\mathbf{p} \cdot \mathbf{f}}{(1 - (\mathbf{u} \cdot \mathbf{t})^2)^{\frac{3}{2}}} \\
&= \frac{\|\mathbf{f}\|\kappa^p}{(1 - (\mathbf{u} \cdot \mathbf{t})^2)^{\frac{3}{2}}} & &= \frac{f\kappa^p}{(1 - (\mathbf{u} \cdot \mathbf{t})^2)^{\frac{3}{2}}} \quad (2)
\end{aligned}
$$

with $f = \|\mathbf{f}\|$ the focal length obtained from camera calibration.

Using the geodesic curvature as the bridge, we can now connect the above two equations to formulate the relationship of the 3D space curve curvature and its perspective projection image curvature:

$$
\frac{\lambda(\mathbf{u} \times \mathbf{T}) \cdot \mathbf{N}}{(1 - (\mathbf{u} \cdot \mathbf{T})^2)^{\frac{3}{2}}}\kappa = \frac{f}{(1 - (\mathbf{u} \cdot \mathbf{t})^2)^{\frac{3}{2}}}\kappa^p \quad (3)
$$

Remark 1. By assuming the osculating plane (spanned by $\mathbf{T}$, $\mathbf{N}$) at the space curve point coincides with the XY plane of the world coordinate system (as in [40]), and using similar techniques as [37], we could also derive the relationship of the 3D space curvature and image curvatures in a projective geometry framework [13, 15]. The world coordinate system and the camera coordinate system could have different orientations, with a homography used to relate the image plane and the osculating plane. We omit the formulas due to space limitation. □

Assuming there are no rotations among the (left, right) camera coordinate systems and the world coordinate system with known calibration, now we can compute the normal $\mathbf{N}$ and curvature κ at a 3D space curve point from image measurements $(x_l, y_l, x_r, y_r, \theta_l, \theta_r, \kappa_l, \kappa_r)$ of one corresponding pair, where f_i is obtained from calibration; $\mathbf{p}_i$, $\mathbf{t}_i$ can be easily computed from the above image measurements, and denote $\mathbf{u}_l = \mathbf{p}_l/\|\mathbf{p}_l\|$, $\mathbf{u}_r = \mathbf{p}_r/\|\mathbf{p}_r\|$; 3D tangent $\mathbf{T}$ and distance (depth) λ_i can also be computed using traditional techniques as described previously, with $i = \{l, r\}$. Thus we have:

Proposition 1: *Given two perspective views of a 3D space curve with full calibration, the normal $\mathbf{N}$ and curvature κ at a space curve point X are uniquely determined from the positions, tangents, and curvatures of its*

projections in two images. Thus the Frenet frame $\{\mathbf{T}, \mathbf{N}, \mathbf{B}\}$ and curvature κ at the space curve point X can be uniquely determined.
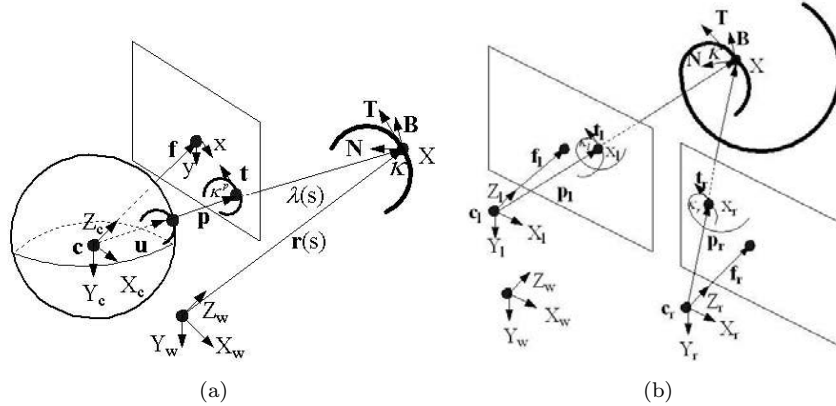

Figure 3:   (a) 3D space curve, its spherical projection to unit sphere, and its perspective projection to image plane. (b) The geometry for Proposition 1. Given $(x_l, y_l, x_r, y_r, \theta_l, \theta_r, \kappa_l, \kappa_r)$, we can compute normal $\mathbf{N}$ and curvature κ at the 3D space curve point. Thus we can determine the position, the Frenet frame $\{\mathbf{T}, \mathbf{N}, \mathbf{B}\}$, and the curvature κ. But the torsion τ can not be determined.

Rewriting eq. (3) to keep the unknown normal $\mathbf{N}$ and curvature κ on the left side, and using it for the left and right images, the relationship of 3D position, tangent, normal, and curvature at a space curve point and its image positions, tangents, and curvatures under perspective projection is:

$$(\mathbf{u}_l \times \mathbf{T}) \cdot \mathbf{N}\kappa = \frac{f_l(1 - (\mathbf{u}_l \cdot \mathbf{T})^2)^{\frac{3}{2}}}{\lambda_l(1 - (\mathbf{u}_l \cdot \mathbf{t}_l)^2)^{\frac{3}{2}}}\kappa_l^p$$

$$(\mathbf{u}_r \times \mathbf{T}) \cdot \mathbf{N}\kappa = \frac{f_r(1 - (\mathbf{u}_r \cdot \mathbf{T})^2)^{\frac{3}{2}}}{\lambda_r(1 - (\mathbf{u}_r \cdot \mathbf{t}_r)^2)^{\frac{3}{2}}}\kappa_r^p \qquad (4)$$

$$\mathbf{T} \cdot \mathbf{N}\kappa = 0$$

where subscript l and r represent measurements from left and right images, respectively. The last equation specifies that $\mathbf{N}$ has to be orthogonal to tangent $\mathbf{T}$. This is a linear system with four unknowns subject to the constraint that $\|\mathbf{N}\| = 1$, which can be treated as three unconstrained unknowns $\mathbf{N}\kappa = \{N_x\kappa, N_y\kappa, N_z\kappa\}$. In general positions this system has a unique solution for these three unknowns, from which we can compute the curvature $\kappa = \|\mathbf{N}\kappa\|$ since $\|\mathbf{N}\| = 1$ and $\kappa > 0$, and normal $\mathbf{N} = \frac{\mathbf{N}\kappa}{\kappa}$. Figure 3 illustrates that from $(x_l, y_l, x_r, y_r, \theta_l, \theta_r, \kappa_l, \kappa_r)$ we can compute the 3D space

curve normal, thus determine the Frenet frame $\{\mathbf{T}, \mathbf{N}, \mathbf{B}\}$ and curvature κ at a 3D space curve point, but not the torsion τ.

Remark 2. While in general configurations there is a unique solution, this system becomes degenerate when the space tangent $\mathbf{T}$ is horizontal (or when $\mathbf{T}$ lies inside the epipolar plane in general epipolar geometry). Then $(\mathbf{u}_l \times \mathbf{T})$ and $(\mathbf{u}_r \times \mathbf{T})$ in the first two equations are parallel (linearly dependent). The space tangent $\mathbf{T}$ cannot be computed and the Frenet frame is undetermined. And also this system has no solution when space curvature κ vanishes. □

Remark 3. The assumption about no rotations (the "rectified" effect) is not essential and rotations can be introduced to the two view calculation as well. □

3   Stereo Correspondence Algorithm

In the previous section, we described how to compute the space curve local approximation given image measurements in two perspective views. Now we describe how this geometric relationship can be used to solve the stereo correspondence problem. The central idea is as follows.

At each point $\alpha(s_0)$ along the space curve, a Frenet approximation can be constructed to characterize the local behavior of the curve *around* that point based solely on the measured information *at* that point. This Frenet approximation supplies geometric constraints on nearby points along $\alpha(s)$. Since measured information is also available for nearby points, it can be compared with the transported information along the *approximation* at $\alpha(s_0)$. Nearby points whose measured information is "close to" the transported information are positively supported, and that which is "far away" are not supported. Both position and orientation information are used in determining these distances. Note that, for a nearby point on another space curve $\beta(s)$, it will be supported by its own neighboring points along β but not by $\alpha(s_0)$.

The situation is analogous to co-circularity in $\mathbb{R}^2$ [30]. Planar curve inference from edge tangents (Fig. 4) indicates how consistent a neighboring tangent $\mathbf{e}_j$ is with a given tangent $\mathbf{e}_i$. For a unit speed planar image curve $\alpha(s)$ with positive curvature, the unique osculating circle that approximates α around s is used to define co-circularity constraints so that nearby (estimated) edge elements can be transported (approximately) along the curve locally. Compatibility functions encode these transport operations. Three distances are used in building them for co-circularity: the transport distance

along the osculating circle, the distance from this point to the edge, and the angular difference between the measured and the transported edges[30, 43]. A relaxation labeling network maximizes a global functional of how well the transported edges match. This selects those tangents that minimize such distances, resulting in geometrically consistent image curve representations. We now seek to do this for space curves in $\mathbb{R}^3$, but the situation is a little more complicated.
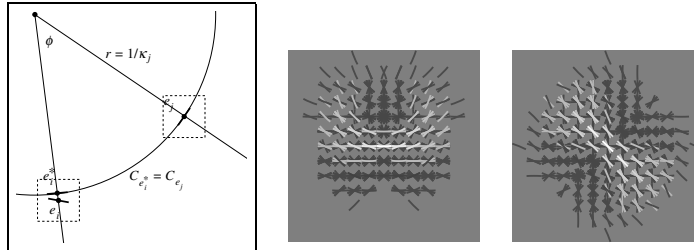

Figure 4:   (LEFT) Co-circularity in the plane indicates how consistent a neighboring tangent $\mathbf{e}_j$ is with a given tangent $\mathbf{e}_i$. Since the positions i and j are close, the actual (edge) curve can be approximated by its osculating circle at j, and curvature is approximately constant. The neighboring tangent $\mathbf{e}_j$ can be transported along the osculating circle, and the mismatch in position, and orientation between $\mathbf{e}_i^*$ and $\mathbf{e}_i$, provides a (distance) measure between them. The larger the mismatch, the bigger the distance. Compatibility functions encode these transport operations. (RIGHT) Examples of the compatibilities around $\mathbf{e}_j$ (center), derived from planar co-circularity. Brightness encodes compatibility value for a neighbor tangent $\mathbf{e}_i$ (of different orientations), with light gray level for high support and dark gray level for low (no) support. Two samples are for $\theta_j = 0°$ and $\kappa_j = -0.2$; and $\theta_j = 45°$ and $\kappa_j = 0.0$, respectively.

3.1   The Discrete Tangent Map

Edges are first detected in each image to provide (x, y, θ, κ), with (x, y) the position coordinates, θ the orientation of the (edge) tangent, and κ the curvature. Special care must be taken at occlusions, junctions, and in general, points where orientation change is not continuous. We allow multiple tangents to be represented at such image points (Fig. 1), again following a biological model [42]. To illustrate: at the image junction in Fig. 5(a), the tangent map has two different tangents, one for the branch in front and one for the branch behind. These two tangents can match with the tangents in the right image along the epipolar line. As a result, both the branch in front and the branch behind have a valid correspondence pair at the junction point.

This multiple-valued representation is not typically used. For feature-based methods that use curve primitives, a standard practice is to first use a traditional edge detector (e.g. Canny [5]), followed by a fitting algorithm

(e.g. B-spline) to the edge chains. But problems arise (see Fig. 5). In our model each well-defined 4-tuple (x, y, θ, κ) is a legitimate entry in the tangent map.
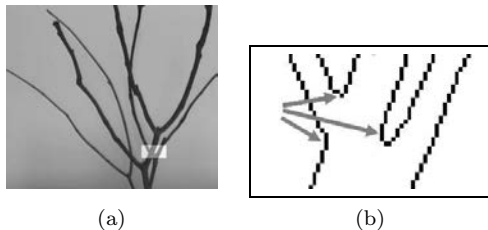

(a) (b)

Figure 5: Problems with standard edge detector: (a) Original image. (b) Edge map of the highlighted region in (a), returned by Canny edge detector. Note that near T-junctions (arrow) Canny edge detector incorrectly smoothes discontinuous orientation changes. This leads to two problems[37]: First, frequently curves are incorrectly joined together (e.g. at T-junctions), thereby smoothing over discontinuous orientation changes (for details see Fig.15 to Fig.17 in [19]). A curve in one image then only matches parts of an incorrectly smooth curve in the other image. Second, point to point correspondence in the image domain is often not unique because of occlusions in space.

To get the discrete tangent map (Fig. 1), we first use logical/linear operators [19] on the original images to get the initial edge tangents and refine these local measurements using planar co-circularity [30] (Fig.4). This coarse orientation selection stage is necessary to eliminate noise and to localize points of singularity; i.e. those initial multiple orientations at one spatial location. It is performed at 16 orientations. Then we use an interpolation function [10] to improve the orientation accuracy and obtain the curvature information. The orientation and spatial quantization is improved by six times. The initial edge tangents are interpolated to 96 quantized orientations and 21 discrete curvatures(Fig. 1). This discrete tangent map is used as the input to our stereo algorithm. The additional accuracy in orientation is required for the orientation disparity component of the compatibility distance function, derived below. Since orientation is quantized around 2π, Light/Dark relations along edges are maintained.

3.2 Geometric Compatibilities in Stereo Tangent Space

The (left,right) tangent maps provide the candidate matches to build the relaxation network. For each item $(x_l, y_l, \theta_l, \kappa_l)$ in the left tangent map, a search along the epipolar line in the right tangent map reveals pairs of possible correspondences. Each stereo tangent pair $(x_l, y_l, x_r, y_r, \theta_l, \theta_r, \kappa_l, \kappa_r)$ is then a node in the *relaxation labeling network* (described in detail later).

The edges that connect the nodes in this network are given by spatial neighborhoods in the image domain. The stereo correspondence problem is then for each node in this network to decide whether it is a correct (TRUE) match or not (FALSE).

To achieve this goal we define a measure of how compatible two nearby nodes are. This measure is derived geometrically through the study of the relationship of the differential properties of 3D objects and their images.

We start with only two (nearby) nodes to illustrate the underlying idea. For each stereo tangent pair, we compute its local space curve Frenet approximation in its osculating plane as described previously. To infer the whole 3D space curve, we transport the measured tangent along its approximation to nearby positions and compare with the measured information there. A relaxation labeling network will select the nodes that are consistent based on their contextual information.

The heart of this process is cartooned in Fig. 6(a), where two space tangents along the 3D Frenet approximation (at one point) are shown. Observe that each of the space tangents projects to a pair of image tangents, so the nodes i and j in the relaxation network consist in *pairs* of image tangents, one in the left image and one in the right, and compatibilities r_{ij} (denoting how compatible i and j are assuming both are correct matches) are defined over these pairs. In particular, while the expected positional disparity between these tangents is introduced, there is also orientation disparity.

A stereo tangent pair, or point in the stereo tangent space, consists in the 8-tuple: $(x_l, y_l, x_r, y_r, \theta_l, \theta_r, \kappa_l, \kappa_r)$. The set of all stereo tangent pairs i in the neighborhood of j, such that i and j are compatible, is called the excitatory *compatibility field* around j.

To build the compatibility fields for stereo, recall that torsion remains a free parameter. [1] However, the Frenet frame does specify the osculating plane in $\mathbb{R}^3$, so we use transport distance along the Frenet approximation in the osculating plane at each point to define the compatibility field (a "piecewise co-circularity" in $\mathbb{R}^3$)(see Fig. 6). Moving along the curve, the Frenet approximation (and the osculating plane) change accordingly. In practice, we found this model works sufficiently well. Since the change of the pose of the osculating plane in $\mathbb{R}^3$ is related to the torsion, the result amounts to a minimal torsion constraint.We conjecture that τ varies smoothly along the curve almost everywhere.

---

[1]From Frenet formulas[12] we have $\mathbf{B}' = -\tau\mathbf{N}$, torsion can be approximated by the finite difference of two nearby nodes in the stereo tangent space, $\tau(s_0) = -\mathbf{B}'(s_0)/\mathbf{N}(s_0) = -\lim_{s_1 \to s_0} \frac{\mathbf{B}(s_1)-\mathbf{B}(s_0)}{(s_1-s_0)\mathbf{N}(s_0)} \approx -\frac{\mathbf{B}(s_1)-\mathbf{B}(s_0)}{(s_1-s_0)\mathbf{N}(s_0)}$. But in practice not much information can be obtained through this computation due to coarse quantization of the measurable quantities.

Given two nearby nodes i and j in the stereo tangent space, which separately encode their own position disparity and orientation disparity in the image pair, the compatibility r_{ij} that both node i and j are "TRUE" (i.e. correct match) is determined by both the position disparity and the orientation disparity as well. We implement it similarly to co-circularity for planar image curves (Fig. 4). Let the reconstructed 3D tangents be $\mathbf{e}_i$, $\mathbf{e}_j$, then project $\mathbf{e}_i$ to the Frenet approximation computed at $\mathbf{e}_j$ in the osculating plane given by the $(\mathbf{T}_j, \mathbf{N}_j)$ frame (Fig. 6). The transported tangent along the approximation to this projection point is $\mathbf{e}_i^*$. We call the distance from $\mathbf{e}_j$ to $\mathbf{e}_i^*$ along the approximation the transport distance s, c the projection distance from $\mathbf{e}_i$ to the approximation, and $\Delta\phi$ the angular rotation between projected $\mathbf{e}_i$ and $\mathbf{e}_i^*$. Then the compatibility r_{ij} that both node i and j are "TRUE" is given by:

$$r_{ij}(TRUE, TRUE) = G_{\sigma_t}(s)G_{\sigma_c}(c)cos(\Delta\phi) \tag{5}$$

where $G_{\sigma_t}(s)$ and $G_{\sigma_c}(c)$ are Gaussian kernels with parameters σ_t, σ_c.
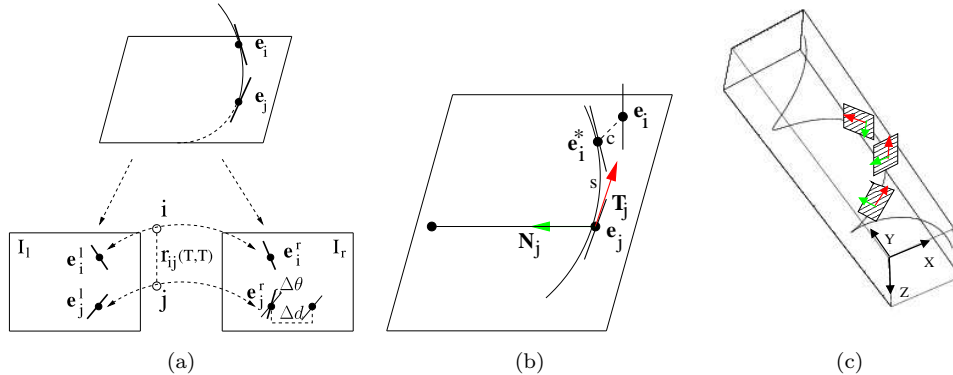

Figure 6: (a) Cartoon of the stereo relaxation process. (top) shows a pair of space tangents associated with the Frenet approximation around the point with tangent $\mathbf{e}_j$. Each of these tangents projects to a (left,right) image tangent pair; compatibility between the space tangents thus corresponds to compatibility over (left,right) image tangent pairs. The projected tangents are shown as thick lines. One left image tangent is redrawn in the right image (as thin lines) to illustrate positional disparity (Δd) and orientation disparity ($\Delta\theta$). (b) Illustration of transported information. The nearby tangent at $\mathbf{e}_i$ is projected onto the Frenet approximation computed at $\mathbf{e}_j$. According to the minimum torsion constraint, this is approximated by projection onto the Frenet approximation at $\mathbf{e}_j$ in the osculating plane given by the $(\mathbf{T}_j, \mathbf{N}_j)$ frame. Thus three components contribute to the compatibility: transport distance along the approximation; projection distance onto the approximation; and angular rotation to tangency. (c) As we move along the space curve, the Frenet approximation (and the osculating plane) change accordingly.

An intuitive demonstration of "slices" of sample compatibility fields r_{ij}

around j are shown in Fig. 7. The left and right images of a 13x13 neighborhood centered at node $j = (940, 0, -940, 0, 131°, 120°, 0.5712, 0.4760)$ (red colored tangent pair in the center) are shown in (a)(b). The position disparity is fixed between the left and right images such that the tangent in the left image matches to the tangent at the same relative position in the right image. With the orientation of the tangents in the right image fixed to be $116°$ (Fig. 7(b)), by varying the orientation of the tangents in the left image at each position, we can show the degree of support a neighboring node i receives from j. Brightess encodes compatibility value with light gray level for high support and dark gray level for low (no) support. Fig. 7(c) shows only the positive support part of Fig. 7(a). Fig. 7(d) shows three slices at different depths, with Fig. 7(c) as the middle one. From these few slices we can see that both position and orientation disparities affect the compatibilities. For display purposes only eight orientations are shown. The computation was performed with a stereo baseline 334mm and focal length 1274 pixels.

Extending this sample compatibility field provides a justification of our choice for the geometric compatibilities. Fig. 8 shows the compatibility fields if we use only image position information, or if we use only position information and image tangents. This figure should be compared with Fig. 7(c). If only the position information is used to determine the compatibility field, the distance from a neighboring node to the central node will be used as the compatibility measure. At each position different orientations have the same magnitude of support without any preference(Fig.8(a)), which is undesirable. If, in addition to the position information we also use image tangents, the compatibility measure will be a combination of distance to the central node and the orientation difference between the two space tangents. The resulting compatibility fields prefer certain orientations (Fig. 8(b)). What is still missing is the "bending" and "twisting" effect (imagine the "bending" and "twisting" of a space curve and the projection of its 3D tangents to the images). The consideration of curvatures is necessary to take this effect into account (see Fig.7(c)). Also keep in mind that this simple comparison is carried out for fixed positional disparities (depth), and the edge tangents in the right image have the same orientation. The difference between these and the proposed compatibility fields becomes more obvious if a 3D version for these is examined (compare with Fig. 7(d)).
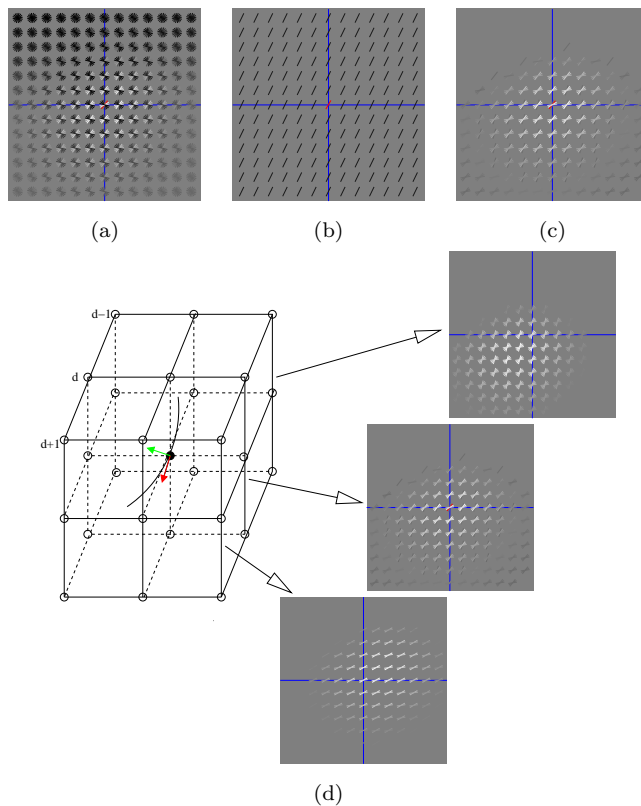
(a)   (b)   (c)

(d)

Figure 7:   Sample compatibility fields around node j (see text for details): (a)(b) Left and right images of a 13x13 neighborhood centered at the two red colored tangents (node j). Fixing the position disparity and the tangent direction in the right image, left image shows different compatibilities from node j for neighboring tangents of different orientations (node i). Brightess encodes compatibility value with light gray level for high support and dark gray level for low (no) support. (c) Redraw (a) to show only the "positive" support part. (d) Slices of compatibility fields at different depth, both position and orientation disparities affect the compatibilities.

3.3   Stereo Via Relaxation Labeling

We now specify the stereo algorithm in more detail. The task is to select pairs of *nodes* in the stereo tangent space that are most consistent with one another by a relaxation labeling network using the discrete compatibility fields just sketched. (They will be refined below.)

Relaxation labeling [18] is a computational framework for finding consistent structures within a network of hypotheses (labels) assigned to nodes in a graph based on the parallel use of local constraints. Suppose a set of nodes are given, and a set of labels are defined for each node. $p_i(\lambda)$ is the probability that label λ is correct for node i, therefore: (1) $0 \leq p_i(\lambda) \leq 1, \forall i, \lambda$; and
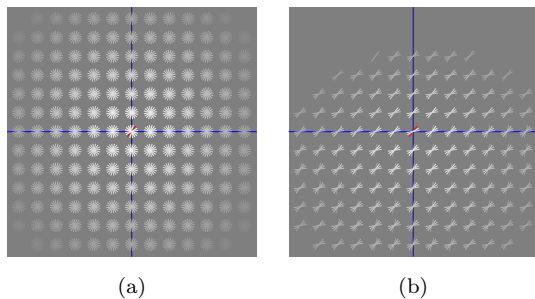
(a)   (b)

Figure 8:   Left image showing alternative compatibility fields: (a) if only position information is used, and (b) if position information plus image tangents are used. Note the lack of "bending" and "twisting" effect, which becomes more obvious if a 3D version is examined.

(2) $\sum_{\lambda \in \Lambda} p_i(\lambda) = 1, \forall i$. The relaxation labeling process starts with a given, often ambiguous labeling assignment and iteratively updates the probabilities to a more consistent one. Consistency between λ for i and λ' for j is denoted by the *compatibilities* $r_{ij}(\lambda, \lambda')$. The *contextual support* label λ on node i receives from its neighbors is: $s_i(\lambda) = \sum_j \sum_{\lambda'} r_{ij}(\lambda, \lambda')p_j(\lambda')$. Labels are selected at each node by an constrained gradient ascent that maximizes the functional $A(p) = \sum_i \sum_\lambda p_i(\lambda)s_i(\lambda) = \sum_i \sum_\lambda \sum_j \sum_{\lambda'} p_i(\lambda)r_{ij}(\lambda, \lambda')p_j(\lambda')$ in parallel for all nodes i and labels λ, using the update rule $p_i^{t+1}(\lambda) = \Pi_{\mathbb{K}}[p_i^t(\lambda) + \delta s_i^t(\lambda)]$, $\Pi_{\mathbb{K}}$ is an operator that projects its argument onto $\mathbb{K}$, the space of all possible assignments, and δ a constant step size.

The choice of relaxation labeling for this computation is motivated by the observation that each decision regarding a stereo tangent pair is coupled with all other possible decisions involving either tangent in the pair coupled in their possible ways. Such coupled decisions are formally games, and relaxation labeling with zero self-payoff terms has been proven to be equivalent to a polymatrix game [27]. Compatibility fields correspond to payoffs, and the consistent points of relaxation labeling are the Nash equilibria of the corresponding game [28]. Computationally such games are examples of linear complementarity problems and other algorithms can be used to solve them [9]. Our compatibilities could also be adopted to other mathematical programming or belief propagation systems; our focus in this paper is on deriving the compatibilities and illustrating their use.

For the stereo correspondence problem, a relaxation network G is built on the *stereo tangent space*. Let $G = (V, E)$, then V is the set of nodes $i = (x_l, y_l, x_r, y_r, \theta_l, \theta_r, \kappa_l, \kappa_r)$ in the *stereo tangent space*. They are the initial stereo correspondence hypotheses, and are obtained by using the epipolar constraint plus thresholds on the maximum allowable position disparity,

16

orientation disparity and curvature disparity. No heuristic constraint (such as ordering constraint) is used. The edges E in this network are determined by the spatial neighborhoods. For node i, an edge is established to another (neighboring) node j which falls into a 13x13x13 region (along x, y, and disparity dimensions, respectively) centered at i, and a geometric compatibility measure denoting how compatible they are will be computed. Each node is assigned two labels $\lambda \in \{T, F\}$, meaning "TRUE" and "FALSE", respectively. Since there are only two possible labels at each node and $p_i(T) + p_i(F) = 1$, we only store $p_i = p_i(T)$ ($p_i(F) = 1 - p_i$). By imposing the design condition that $s_i(F) = -s_i(T)$, we can realize the compatibility structure to be [3]: $r_{ij}(T,T) = -r_{ij}(T,F) = -r_{ij}(F,T) = r_{ij}(F,F)$. Thus we can simplify notation with p_i, r_{ij}, and s_i, referring to $p_i(T)$, $r_{ij}(T,T)$, and $s_i(T)$, respectively. Then the update rule takes a very simple formula:

$$p_i^{t+1} = \Pi_0^1[p_i^t + \delta \sum_j r_{ij} p_j^t] \qquad (6)$$

where $\Pi_0^1(x) \triangleq min(1, max(0, x))$, denoting projection onto [0,1], and δ is a constant step size. In our stereo system we set the confidences of all initial hypotheses with $p_i^0 = 0.5$, and iteratively update the confidences with the above update rule, using the geometric compatibilities r_{ij} as described previously. Initial hypotheses that are compatible with the context in which they are embedded get enough support to incerease their probabilities. After the relaxation process converges, a threshold ($p \geq 0.95$ in our experiments) reveals all (TRUE) correspondences.

3.4 Refining Compatibility Fields

Even with the local geometric relationship derived above, however, there still can be ambiguities along the epipolar line; we now address these in the remaining two terms necessary for our compatibility functions. As we show, both inhibit putative reconstructions that stand improperly in depth. To illustrate, Fig. 9 shows the reconstruction based solely on geometric relationships. In the front view (Fig. 9(c)), the left and right parts have similar (correct) depth, while the middle parts are false matches. The rotated view (Fig. 9(d)) shows the correct matches as the middle two branches with similar depth. Such false matches will generically arise from the geometry alone because one point in the left image could possibly match to multiple points in the right image.

A. Disparity Gradient Limit and Ordering Constraints Revisited: Different constraints on nearby matching pairs have been proposed: *The*
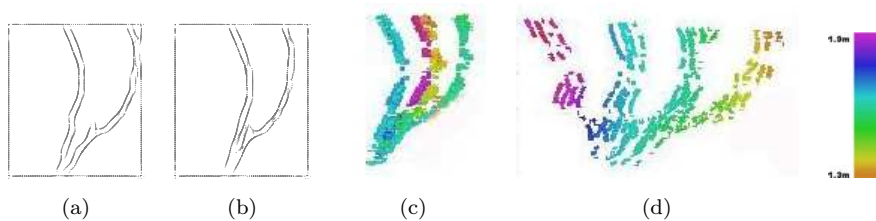
Figure 9: False matches from attempting correspondence on a pair of twigs, because the geometric compatibilities can apply to all possible matches. (e.g., the left branch in (a) can match both the left and right brances in (b), and so on.) (a)(b) Left and right tangent maps. (c) Front view of reconstruction showing false matches (the middle two). (d) Rotated view showing false matches plus ghosts from noise in the tangent map. Depth scale shown at right.

ordering constraint holds that it is geometrically impossible for points arising from the same opaque surface to be differently ordered in the two images [33, 13]; *The disparity gradient limit constraint* states that, given camera geometry approximating the arrangement of human eyes, a disparity gradient limit of 1 for points on a single object will almost always be satisfied for the correct matches [33]. Much psychophysical and analytical study supports this.

However, as remarked in Sec.1, these constraints frequently break down for the examples in this article. The discrepency arises from ubiquitously (but incorrectly) assuming that, if only a small neighborhood is used in matching, then every primitive inside this region comes from a single object. If this is the case, then such constraints can be applied; if not, they cannot. A close look at the problem regions we pinpointed previously reveals this point immediately: for the image primitives *from a single object* (e.g. either the front branch or the back branch in the highlighted region of Fig.1), these constraints hold. It is not these constraints but their use with incorrect assumptions that is misleading: Small image neighborhood does not imply single object.

A chicken-and-egg problem emerges: our task for stereo correspondence must determine whether a candidate matching pair is a true match or not, i.e. whether it comes from an object in 3D space. But without knowing this, how can we determine which other neighboring pairs come from the same object so that the above constraints can be safely applied? For curves the Frenet approximation extends image position: In general, neighbors whose tangents continue (in 3D space) likely come from the same object.

We exploit this as follows. For a matching pair, if its computed 3D tangent deviates greatly from the frontal parallel plane, then it will propagate space curves with large disparity gradient values. For such pairs we sim-

18

ply suppress their geometric compatibilities. The new compatibility value is given by:

$$r'_{ij} = L(\mathbf{T}_j)r_{ij} \qquad\qquad (7)$$

where r_{ij} is computed as in the previous subsection and $L(\mathbf{T}_j) = \{ \begin{array}{ll} 0, & DG(\mathbf{T}_j) > 1 \\ 1, & otherwise \end{array}$,
with $DG(\mathbf{T}_j)$ the Disparity Gradient for space tangent $\mathbf{T}_j$ at position given by node j.

B. Near/Far Inhibition: Our final modification of the compatibilities implements the uniqueness constraint often applied in the image domain, but again lifted into space. It functions in effect as a prior for the geometry, inhibiting matches "near" and "far" from strong, preferred matches. Matches are preferred in decreasing order from fixation and in proportion to geometric match support(Fig. 10). We observe there is a direct biological realization of this prior. We remarked earlier that binocular neurons were orientation selective. We now observe that they also can be classified into three groups [31](Fig. 10): (1)*Tuned excitatory neurons* form a group that are disparity selective over a limited (and often narrow) range. (2)*Far neurons* exhibit a selectivity for uncrossed disparities; and (3)*Near neurons* are selective for crossed disparities. Far and near neurons are complementary: One set gives excitatory responses to objects farther than the point of fixation and inhibitory responses to nearer objects; while other set has the opposite behavior, excitation for nearer objects and inhibition for further ones [22][2].

4 Experimental Results

4.1 Synthetic Example

We first assessed the accuracy of our algorithm with synthetic images. Twenty algebraic 3D space curves were generated with parameters chosen randomly. In these are five helices, five elliptical helices, and ten piecewise polynomial 3D space curves (each obtained using cubic spline interpolation with ten control points). The curves were all started at a distance of 1800mm, and

---

[2]We believe this explains the double-nail illusion [21]. The configuration consists of two "nails" held straight ahead at reading distance, one a few centimeters behind the other at the same eye level. The viewer perceives two illusory nails side by side rather than one in front of the other. This follows because they are in the disparity tuning zone of the tuned excitatory neurons, (that is, consistent with our geometry), and because the inhibition of the near and far neurons, the (true) front and back nails are not perceived.
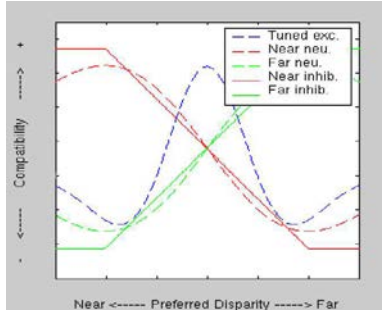
Figure 10: Near and Far neuron tuning function. In biological terms depth can be encoded by neurons tuned to specific disparities; this is cartooned here as an excitatory neuron tuned to 0 disparity. Also shown are tuning curves for near and far neurons (in dashed lines), which are excited (resp. inhibited) by stimuli closer (resp. further) than the point of fixation. In our implementation (in solid lines), the geometric compatibility functions are refined by these tuning curves. Thus potential matches are supported by geometric consistency but inhibited by better matches in the neighborhood.

ranged in depth from about 1640mm to 1960mm. These were then projected through the camera model to create an image pair, and our algorithm was run on this pair. The results were then compared with ground truth (Fig. 11). The quantization in the system is such that one pixel corresponds to about 9.3mm (at 1.8m), which shows that our edge detection and interpolation are functioning to sub-pixel accuracy.
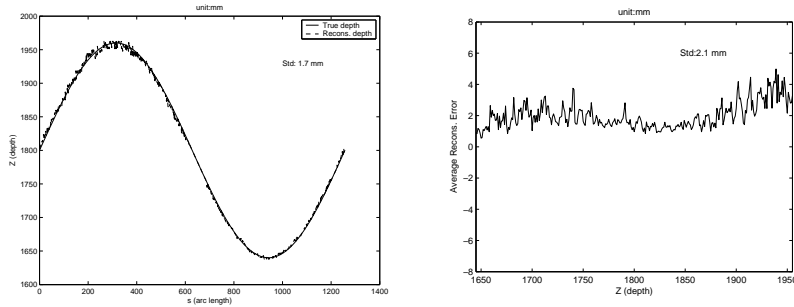

Figure 11: Error analysis from a series of 20 algebraic curves with random parameters (see text). (LEFT) Plot of reconstructed depth and true depth vs. arc length for an elliptical helix; standard deviation: 1.7mm; maximal difference 8.0mm. (RIGHT) Average depth reconstruction error (absolute value) of 20 3D curves vs. depth. Standard deviation: 2.1mm.

20

4.2   Natural Scene Examples

Fig. 12 shows the 3D reconstruction of image pair (Fig. 1) by our algorithm. The color depth map indicates both the front and the back branches are correctly reconstructed. As pointed out earlier, at the image junction point in the left image the tangent map has two different tangents, one for the branch in front and one for the branch behind. These two tangents both get matched with the tangents in the right image along the epipolar line. As a result, both the branch in front and the branch behind have a valid correspondence pair at the junction point and they are both recovered as smooth space curves. Such reconstructions can be observed throughout the experimental section. A few false matches (about 0.9%) between the two layers can be seen from the birds-eye view. The running time of our stereo algorithm on this pair is 43 seconds on a Intel Xeon 2.4GHz PC with 1GB memory. The computation of the geometric compatibility fields accounts for about 70% of the total computation. And the relaxation process converged after 5 iterations. A pair of Sony XC-555 cameras equiped with 12mm lenses were used for the experiments. The baseline for this stereo pair is 128mm. Parameters for the geometric compatibilities (eq. (5)) are $\sigma_t = 3.0mm$, $\sigma_c = 1.5mm$ with the selection based on the quantization of our system and the size of the neighborhood region that we use. Parameter values and number of iterations remain fixed for all experiments in this section.

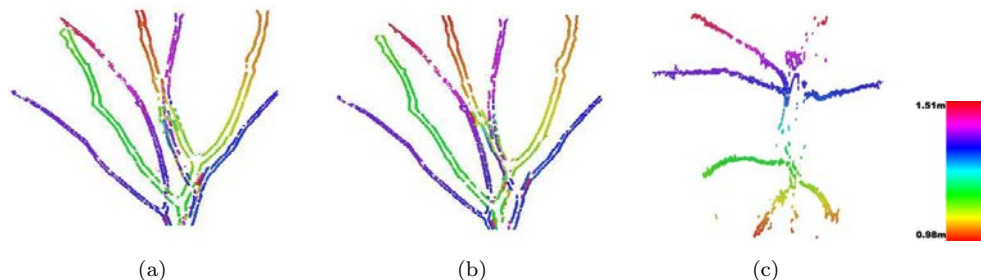(a)   (b)   (c)

Figure 12:   3D reconstruction of Fig. 1: (a)(b) Results at two different viewpoints. (c) Birds-eye view. Note the two layers are clearly separated, and the false matches in (c) form short segments rather than long curves and could be easily removed. Colored depth scale is shown at right (units: meters).

Fig. 13 shows a more complex scene which is a lamp with curves defining its circular shape. The space configuration of this lamp is that curves winding from upper left to lower right are closer than the vertical bar, while curves winding from upper right to lower left are further away than the

vertical bar. Fig. 14 shows our reconstruction result. The 3D configuration is correctly recovered when compared to the depth scale at right [3]. The running time of our stereo algorithm on this pair is 84 seconds. The gaps between curve segments arise for two reasons in our reconstruction. First, edge information is missing from some crossings, so no reconstruction is available for these short segments. Second, when image edge tangents are horizontal (or when they pass epipoles in general stereo configuration), the solution to the 3D tangent is unstable and the 3D Frenet frame can not be determined (Remark 2). In our algorithm these edge tangents are not used for reconstruction, which explains the lack of 3D reconstruction for horizontal edges near the top of lamp pair.

Since no other feature-based algorithms are available for comparison, we experimented with several dense stereo algorithms [36]. Maintaining only the confident values ($c > 0.04$) of the cooperative algorithm [41] provided the best results among these algorithms (Fig. 14(c)) for this pair. Note the missing curves at several places and the lack of smooth depth change along the curves. As we remarked earlier, although their algorithm also uses neighboring support between nearby matching pairs, our geometric consistency requirement outperforms traditional methods for these difficult scenes.

A natural plant example is shown in Fig. 15, where the matched curve points are overlayed with the orginal image pair to provide an intuitive idea of which curve points are matched.

In Fig. 16 we show an example of wires winding in a complex fashion relevant to complex grouping tasks in 3D. The three layers are clearly separated and their respective depth is vivid when compared to the depth scale at right. For this pair only the curve points on the wires are used to test our algorithm. Note that when space curvature κ vanishes the Frenet frame is undetermined. Under this situation our system degrades gracefully where the space curve local approximation is essentially the tangent line.

To compare with other curve based algorithms, we emulate the experiment performed in [37]. Our result(Fig. 17) is shown with matched curves overlayed on the original images. Note that although they also use image curvatures, the fundamental difference between our algorithm and theirs is that we use "transport", or the moving Frenet frames to geometrically connect neighboring matching pairs. This provides extra geometric constraint. Finally in Fig. 18 is another natural scene example. These last two examples

---

[3]All image pairs and results (in this paper) are available at http://www.cs.yale.edu/˜ligang/research/CurveStereo
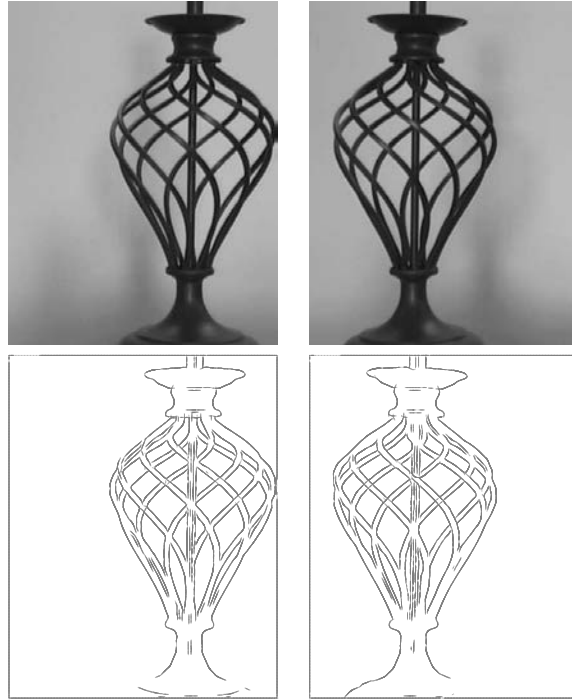
Figure 13: (TOP) Left and right images of a lamp. The space configuration of this lamp is that curves winding from upper left to lower right are closer than the vertical bar, while curves winding from upper right to lower left are further away than the vertical bar. (BOTTOM) Discrete tangent map.

have a relatively "planar" structure.

5   Summary

By relating the differential structures at image points (position, tangent, and curvature) in the left and right images with the geometry of the space curve point (position, Frenet frame, and curvature), we show how nearby image correspondence pairs should be consistent with the Frenet approximation of the space curve, and how this compatibility relation can be put into a stereo relaxation labeling network where positional disparity and orientation disparity are combined naturally. Examples show how well our algorithm performs on scenes with complex 3D structure, especially at places where heuristic constraints fail.

Although there was not sufficient space to develop the neurobiological foundations of our algorithm, we observe in closing how important they were
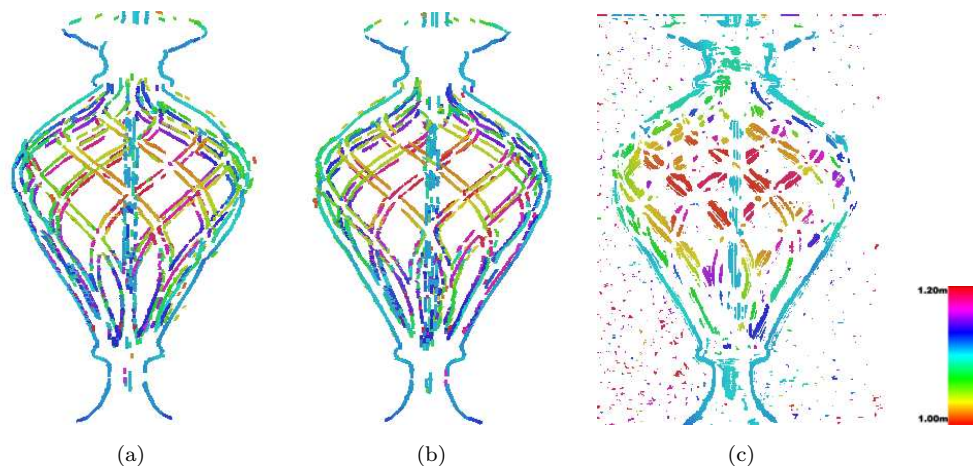
Figure 14: 3D Reconstruction of the lamp pair: (a)(b) Our results at two different viewpoints. (c) Cooperative algorithm result [41]. Colored depth scale is shown at right (units: meters).

to its formulation. Continuing along this thought suggests that the next step in developing these algorithms will invoke feedback from the reconstruction to the initial edge maps, analagous to the feedback between different visual areas, to complete those curves poorly inferred from monocular data.

References

[1] Shamez Alibhai and Steven W. Zucker. Contour-based correspondence for stereo. In *Proc. of ECCV*, 2000.

[2] Stephen T. Barnard and Martin A. Fischler. Computational stereo. *ACM Computing Surveys*, 14(4):553–572, 1982.

[3] Ohad Ben-Shahar and Steven W. Zucker. The perceptual organization of texture flow: A contextual inference approach. *IEEE Trans. on PAMI*, 25(4):401–417, 2003.

[4] Myron Z. Brown, Darius Bruschka, and Gregory D. Hager. Advances in computational stereo. *IEEE Trans. on PAMI*, 25(8):993–1008, 2003.

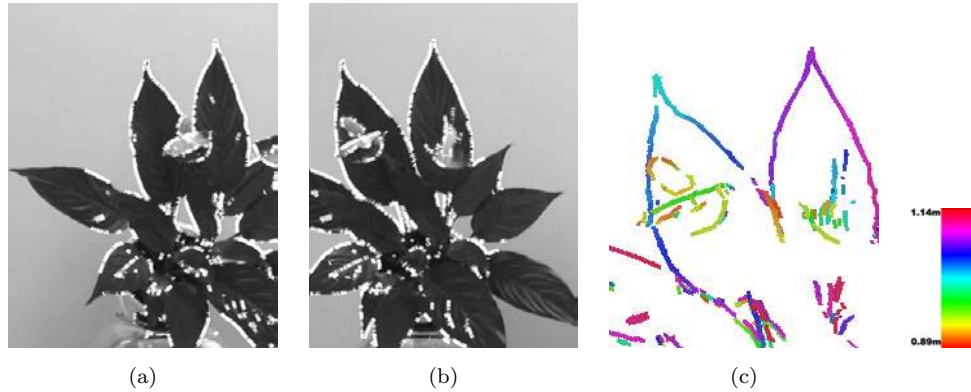[5] John Canny. A computational approach to edge detection. *IEEE Trans. on PAMI*, 8(6):679–698, 1986.

Figure 15: 3D reconstruction of the plant pair: (a)(b) Matched curve points overlayed with the original image pair. (c) Zoom in view of the upper left part. Colored depth scale is shown at right (units: meters).
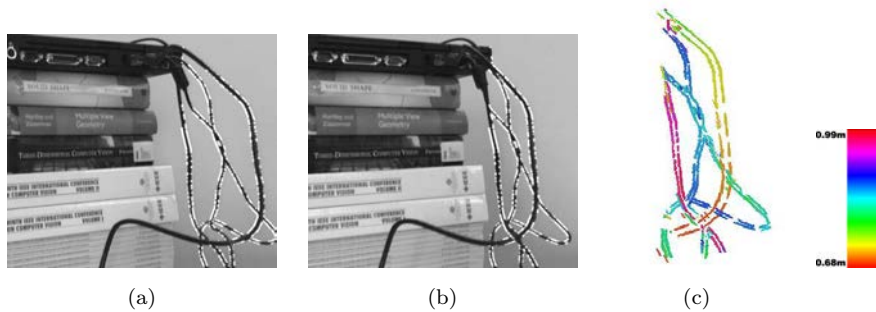

Figure 16: (a)(b) Matched curve points on the wires overlayed with the original image pair. (c) 3D reconstruction of the wires. Note the three layers are clearly separated. Colored depth scale is shown at right (units: meters).

[6] William J. Christmas, Josef Kittler, and Maria Petrou. Structural matching in computer vision using probabilistic relaxation. *IEEE Trans. on PAMI*, 17(8):749–764, 1995.

[7] Roberto Cipolla and Peter Giblin. *Visual Motion of Curves and Surfaces*. Cambridge Univ. Press, 2000.

[8] Roberto Cipolla and Andrew Zisserman. Qualitative surface shape from deformation of image curves. *International Journal of Computer Vision*, 8:53–69, 1992.

[9] Richard W. Cottle, Jong-Shi Pang, and Richard E. Stone. *The Linear Complementarity Problem*. Academic Press, 1992.
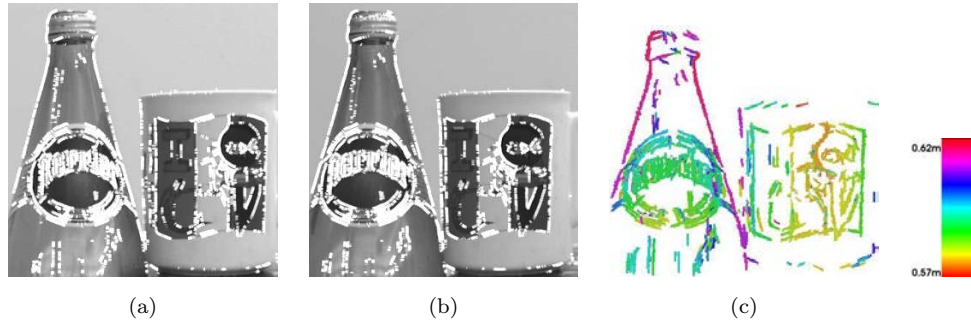
Figure 17: (a)(b) Matched curves overlayed with original image pair. (c) 3D reconstruction. Colored depth scale is shown at right (units: meters).
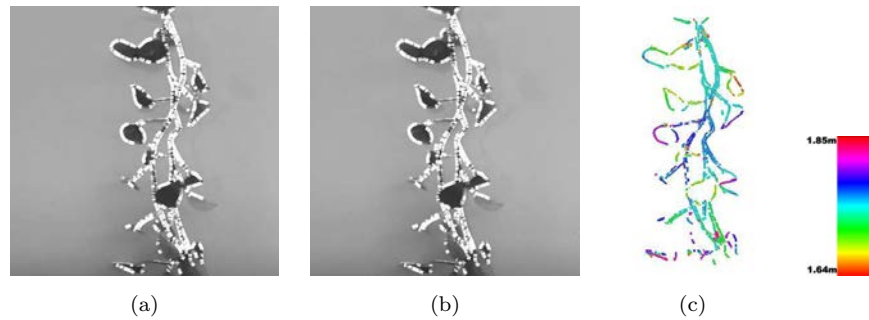

Figure 18: (a)(b) Matched curve points overlayed with the original image pair. (c) 3D reconstruction. Colored depth scale is shown at right (units: meters).

[10] Chantal David and Steven W. Zucker. Potentials, valleys, and dynamic global coverings. *International Journal of Computer Vision*, 5:219–238, 1990.

[11] Umesh R. Dhond and J. K. Aggarwal. Structure from stereo – a review. *IEEE Trans. on Systems, Man, and Cybernetics*, 19(6):1489–1510, 1989.

[12] Manfredo P. do Carmo. *Differential Geometry of Curves and Surfaces.* Prentice-Hall, Inc., 1976.

[13] Olivier Faugeras. *Three-Dimensional Computer Vision.* The MIT Press, 1993.

[14] Olivier Faugeras and Luc Robert. What can two images tell us about a third one? *International Journal of Computer Vision*, 18:5–19, 1996.

[15] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge Univ. Press, 2000.

[16] Ian P. Howard and Brian J. Rogers. *Binocular Vision and Stereopsis*. Oxford Univ. Press, 1995.

[17] D. H. Hubel and T. N. Wiesel. Functional architecture of macaque monkey visual cortex. *Proc. R. Soc. Lond. B.*, 198:1–59, 1977.

[18] Robert A. Hummel and Steven W. Zucker. On the foundations of relaxation labeling processes. *IEEE Trans. on PAMI*, 5(3):267–287, 1983.

[19] Lee A. Iverson and Steven W. Zucker. Logical/linear operators for image curves. *IEEE Trans. on PAMI*, 17(10):982–996, 1995.

[20] David G. Jones and Jitendra Malik. Determining three-dimensional shape from orientation and spatial frequency disparities. In *Proc. of ECCV*, 1992.

[21] Jodi D Krol and Wim A van de Grind. The double-nail illusion: Experiments on binocular vision with nails, needles, and pins. *Perception*, 9:651–669, 1980.

[22] S. Lehky and T. Sejnowski. Neural model of stereoacuity and depth interpolation based on a distributed representation of stereo disparity. *J. Neurosci.*, 10:2281–99, 1990.

[23] David Marr. *Vision*. W.H. Freeman and Company, 1982.

[24] David Marr and Tomaso Poggio. Cooperative computation of stereo disparity. *Science*, 194:283–287, 1976.

[25] David Marr and Tomaso Poggio. A computational theory of human stereo vision. *Proc. Royal Soc. London B,*, 204:301–328, 1979.

[26] G.G. Medioni and R. Nevatia. Segment-based stereo matching. *CVGIP*, 31(1):2–18, July 1985.

[27] Douglas A. Miller and Steven W. Zucker. Copositive-plus lemke algorithm solves polymatrix games. *Operations Research Letters*, 10:285–290, 1991.

[28] Douglas A. Miller and Steven W. Zucker. Efficient simplex-like methods for equilibria of nonsymmetric analog networks. *Neural Computation*, 4:167–190, 1992.

[29] Nasser M. Nasrabadi. A stereo vision technique using curve-segments and relaxation matching. *IEEE Trans. on PAMI*, 14(5):566–572, 1992.

[30] Pierre Parent and Steven W. Zucker. Trace inference, curvature consistency, and curve detection. *IEEE Trans. on PAMI*, 11(8):823–839, 1989.

[31] G. F. Poggio and B. Fischer. Binocular interaction and depth sensitivity in striate and prestriate cortex of behaving rhesus monkey. *J. Neurophys.*, 40:1392–1405, 1977.

[32] Gian F. Poggio and Tomaso Poggio. The analysis of stereopsis. *Annual Review of Neuroscience*, 7:379–412, 1984.

[33] Stephen B Pollard, John E W Mayhew, and John P Frisby. Pmf: A stereo correspondence algorithm using a disparity gradient limit. *Perception*, 14:449–470, 1985.

[34] Alison F. Richard. *Primates in Nature*. W.H. Freeman and Company, 1985.

[35] Luc Robert and Olivier Faugeras. Curve-based stereo: Figural continuity and curvature. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1991.

[36] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1/2/3):7–42, 2002.

[37] Cordelia Schmid and Andrew Zisserman. The geometry and matching of lines and curves over multiple views. *International Journal of Computer Vision*, 40(3):199–233, 2000.

[38] Ying Shan and Zhengyou Zhang. New measurements and corner-guidance for curve matching with probabilistic relaxation. *International Journal of Computer Vision*, 46(2):157–171, 2002.

[39] Richard P. Wildes. Direct recovery of three-diemensional scene geometry from binocular stereo disparity. *IEEE Trans. on PAMI*, 13(8):761–774, 1991.

[40] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Trans. on PAMI*, 22(11):1330–1334, 2000.

[41] Charles Zitnick and Takeo Kanade. A cooperative algorithm for stereo mathching and occlusion detection. *IEEE Trans. on PAMI*, 22(7):675–684, 2000.

[42] Steven W. Zucker. Which computation runs in visual cortical columns? In J. Leo van Hemmen and T.J. Sejnowski, editors, *Problems in Systems Neuroscience*. Oxford University Press, In Press, 2004.

[43] Steven W. Zucker, Allan Dobbins, and Lee Iverson. Two stages of curve detection suggest two styles of visual computation. *Neural Computation*, 1:68–81, 1989.